

Commitment Under Uncertainty: Two-Stage Stochastic Matching Problems

Irit Katriel*, Claire Kenyon-Mathieu, and Eli Upfal**

Brown University
{irit,claire,eli}@cs.brown.edu

Abstract. We define and study two versions of the bipartite matching problem in the framework of two-stage stochastic optimization with recourse. In one version the uncertainty is in the second stage costs of the edges, in the other version the uncertainty is in the set of vertices that needs to be matched. We prove lower bounds, and analyze efficient strategies for both cases. These problems model real-life stochastic integral planning problems such as commodity trading, reservation systems and scheduling under uncertainty.

1 Introduction

Two-stage stochastic optimization with recourse is a popular model for hedging against uncertainty. Typically, part of the input to the problem is only known probabilistically in the first stage, when decisions have a low cost. In the second stage, the actual input is known but the costs of the decisions are higher. We then face a delicate tradeoff between speculating at a low cost vs. waiting for the uncertainty to be resolved.

This model has been studied extensively for problems that can be modeled by linear programming (sometimes using techniques such as Sample Average Approximation (SAA) when the linear program (LP) is too large.) Recently there has been a growing interest in 2-stage stochastic combinatorial optimization problems [1, 2, 6, 12, 19–22, 24]. Since an LP relaxation does not guarantee an integer solution in general, one can either try to find an efficient rounding technique [11] or develop a purely combinatorial approach [5, 8]. In order to develop successful algorithmic paradigms in this setting, there is an ongoing research program focusing on classical combinatorial optimization problems [23]: set cover, minimum spanning tree, Steiner tree, maximum weight matching, facility location, bin packing, multicommodity flow, minimum multicut, knapsack, and others. In this paper, we aim to enrich this research program by adding a basic combinatorial optimization problem to the list: the minimum cost maximum bipartite matching problem. The task is to buy edges of a bipartite graph

* Supported in part by NSF awards DMI-0600384 and ONR Award N000140610607.

** Supported in part by NSF awards CCR-0121154 and DMI-0600384, and ONR Award N000140610607.

which together contain a maximum-cardinality matching in the graph. We examine two variants of this problem. In the first, the uncertainty is in the second stage edge-costs, that is, the cost of an edge can either grow or shrink in the second stage. In the second variant, all edges become more expensive in the second stage, but the set of nodes that need to be matched is not known.

Here are some features of minimum cost maximum bipartite matching that make this problem particularly interesting. First, it is not subadditive: the union of two feasible solutions is not necessarily a solution for the union of the two instances. In contrast, most previous work focused on subadditive structures, with the notable exception of Gupta and Pál’s work on stochastic Steiner Tree [9]. Second, the solutions to two partial instances may interfere with one another in a way that seems to preclude the possibility of applying cost-sharing techniques associated with the scenario-sampling based algorithms [9, 10]. This intuitively makes the problem resistant to routine attempts, and indeed, we confirm this intuition by proving a lower bound which is stronger than what is known¹ for the sub-additive problems: in Theorem 5, we prove a hardness of approximation result in the setting where the second-stage scenarios are generated by choosing vertices independently. It is therefore natural that our algorithms yield upper bounds which are either rather weak (Theorem 2, Part 1) or quite specialized (Theorem 7). To address this issue, we relax the constraint that the output be a maximum matching, and consider bicriteria results, where there is a tradeoff between the cost of the edges bought and the size of the resulting matching (Theorem 2, Part 2, and Theorem 8). This approach may be a way to circumvent hardness for other stochastic optimization problems as well.

Although the primary focus of this work is stochastic optimization, another popular objective for the prudent investor is to minimize, not just the expected future cost, but the maximum future cost, over all possible future scenarios: that is the goal of robust optimization. We also prove a bicriteria result for robust optimization (Theorem 3.) Guarding oneself against the worst case is more delicate than just working with expectations. The solution requires a different idea: preventing undesirable high-variance events by explicitly deciding, against the advice of the LP solution, to not buy expensive edges (To analyze this, the proof of Theorem 3 involves some careful rounding.) This general idea might be applicable to other problems as well.

We note that within two-stage stochastic optimization with recourse, matching has been studied before [15]. However, the problem studied here is very different: there, the goal was to construct a maximum weight matching instead of the competing objective of large size and small cost; moreover the set of edges bought by the algorithm had to form exactly a matching instead of just contain a matching. In Figure 1, we give an example illustrating the difference between requiring equality with a matching or containment of a matching.

¹ To the best of our knowledge, all previous hardness results hold only when the second stage scenarios are given explicitly, i.e., when only certain combinations of parameter settings are possible.

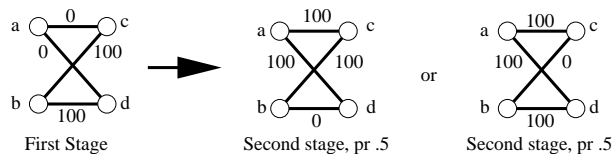


Fig. 1. An example in which buying edges speculatively can help.

Our main goal in this paper is to further fundamental understanding of the theory of stochastic optimization; however, we note that a conceivable application of this problem is commodity transactions, which can be viewed as a matching between supply and demand. When the commodity is indivisible, the set of possible transactions can be modeled as a weighted bipartite graph matching problem, where the weight of an edge represents the cost or profit of that transaction (including transportation cost when applicable). A trader tries to maximize profits or minimize total costs depending on her position in the transaction. A further tool that a commodity trader may employ to improve her income is timing the transaction. We model timing as a two-stage stochastic optimization problem with recourse: The trader can limit her risk by buying an option for a transaction at current information, or she can assume the risk and defer decisions to the second stage. Two common uncertainties in commodity transactions, price uncertainty and supply and demand uncertainty, correspond to the two stochastic two-stage matching problems mentioned above: finding minimum weight maximum matching with uncertain edge costs, and finding maximum matching with uncertain matching vertices. Similar decision scenarios involving matchings also show up in a variety of other applications such as scheduling and reservation systems.

Our results are summarized in the following table. We first prove (Theorem 1) that, with explicit scenarios, the uncertain matching vertices case is in fact a special case of the uncertain edge costs case. Then, it suffices to prove upper bounds for the more general variant and lower bounds for the restricted one. For the problem of minimizing the expected cost of the solution, we show an approximability lower bound of $\Omega(\log n)$. We then describe an algorithm that finds a maximum matching in the graph at a cost which is an n^2 -approximation for the optimum. We then show that by relaxing the demand that the algorithm constructs a maximum matching, we can “beat” the lower bound: At a cost of at most $1/\beta$ times the optimum, we can match at least $n(1 - \beta)$ vertices. Furthermore, we show that a similar bicriteria result holds also for the robust version of the problem, i.e., when we wish to minimize the worst-case cost.

With independent choices in the second-stage scenarios, our main contribution is the lower bound. The reduction of Theorem 1 does not apply, but we prove, for for both types of uncertainty, that it is NP-hard to approximate the problem within better than a certain constant factor. We also prove an upper bound for a special case of the uncertain matching vertices variant.

Input:	Explicit Scenarios		Independent Choices
Criteria:	Expected Cost	Worst-Case Cost	Expected Cost
Uncertain edge costs	<ul style="list-style-type: none"> • n^2-approximation of the cost to get a maximum matching [Theorem 2, part 1] • $1/\beta$-approximation of the cost to match at least $n(1 - \beta)$ vertices [Theorem 2, part 2] <ul style="list-style-type: none"> • Same hardness results as below [Theorem 1] 	<p>factor $1/\beta$ approximation of the cost to match at least $n(1 - \beta)$ vertices [Theorem 3]</p>	<p>NP-hard to approximate within a certain constant [Theorem 6]</p>
Uncertain matching vertices	<ul style="list-style-type: none"> • $\Omega(\log n)$ approximability lower bound [Theorem 4, Part 1] • NP-hard already for two scenarios [Theorem 4, Part 2] • Same upper bounds as above [Theorem 1] 	<p>As above [Theorem 1]</p>	<ul style="list-style-type: none"> • As above [Theorem 5] • approximation for a special case [Theorem 7]

2 Explicit scenarios

In this section, we assume that we have an explicit list of possible scenarios for the second stage.

Uncertain edge costs. Given a bipartite graph $G = (A, B, E)$, we can buy edge e in the first stage at cost $C_e \geq 0$, or we can buy it in the second stage at cost $C_e^s \geq 0$ determined by the scenario s . The input has an explicit list of scenarios, and known edge costs (c_e^s) in scenario s . For uncertain edge costs, without loss of generality we can assume that $|A| = |B| = n$ and that G has a perfect matching (PM). Indeed, there is an easy reduction from the case where the maximum matching has size k : just create a new graph by adding a set A' of $n - k$ vertices on the left side, a set B' of $n - k$ vertices on the right side, and edges between all vertex pairs in $A' \times B$ and in $A \times B'$, with cost 0.

In the *stochastic optimization* setting, the algorithm also has a known second stage distribution: scenario s occurs with probability $\Pr(s)$. The goal is, in time polynomial in both the size of the graph and the number of scenarios, to minimize the *expected* cost; if E_1 denotes the set of edges bought in the first stage and E_2^s the set of edges bought in the second stage under scenario s , then:

$$\text{OPT}_1 = \min_{E_1, E_2^s} \left\{ \sum_{s \in S} \Pr(s) \left(\sum_{e \in E_1} C_e + \sum_{e \in E_2^s} C_e^s \right) : \forall s, E_1 \cup E_2^s \text{ has a PM} \right\} \quad (1)$$

Stochastic optimization with uncertain edge costs has been studied for many problems, see for example [10, 17].

In the *robust optimization* setting, the goal is to minimize the maximum cost (instead of the expected cost):

$$\text{OPT}_2 = \min_{E_1, E_2^s} \left\{ \max_{s \in S} \left(\sum_{e \in E_1} C_e + \sum_{e \in E_2^s} C_e^s \right) : \forall s, E_1 \cup E_2^s \text{ has a PM} \right\} \quad (2)$$

Robust optimization with uncertain edge costs has also been studied for many problems, see for example [4].

Uncertain activated vertices. In this variant of the problem, there is a known distribution over scenarios s , each being defined by a set $B_s \subset B$ of *active* vertices that are allowed to be matched in that scenario. Each edge costs c_e today (before B_s is known) and τc_e tomorrow, where $\tau > 1$ is the *inflation parameter*. As in Expression 1, the goal is to minimize the expected cost, i.e.,

$$\text{OPT}_3 = \left\{ C(E_1) + \tau \sum_{s \in S} \Pr(s) C(E_2^s) : \forall s, E_1 \cup E_2^s \text{ contains max matching of } (A, B_s, E \cap (A \times B_s)) \right\} \quad (3)$$

Stochastic optimization with uncertain activated vertices has also been previously studied for many problems, see for example [9]. There is a similar expression for robust optimization with uncertain activated vertices.

Theorem 1 (Reduction). *The two-stage stochastic matching problem with uncertain activated vertices and explicit second-stage scenarios (OPT_3) reduces to the case of uncertain edge costs and explicit second-stage scenarios (OPT_1).*

Proof. We give an approximation preserving reduction. Given an instance with stochastic matching vertices, we transform it to an instance of the problem with stochastic edge-costs, as follows. Assume that our input graph is $G = (A, B, E)$ where $A = \{a_1, \dots, a_{|A|}\}$ and $B = \{b_1, \dots, b_{|B|}\}$. We first add a set $A' = \{a'_1, \dots, a'_{|B|}\}$ of $|B|$ new vertices to A , and connect each a'_i with b_i by an edge. In other words, we generate the graph $G' = (A \cup A', B, E \cup \{(a'_i, b_i) : 1 \leq i \leq |B|\})$.

For the edges between A and B , edge costs are the same as in the original instance, in the first stage as well as the second stage. The costs on the edges between A' and B create the effect of selecting the activated vertices: For each (a'_i, b_i) , the first-stage cost is n^2W , and the second-stage cost is n^2W if b_i is active and 0 otherwise. Here, W is the maximum cost of an edge, nW is an upper bound on the cost of the optimal solution, and n^2W is large enough that any solution containing this edge cannot be an optimal, or even an n -approximate solution. Hence, a second-stage cost of 0 for (a'_i, b_i) allows b_i to be matched with a'_i for free, while a cost of nW forces b_i to be matched with a vertex from A . This concludes the reduction. \square

From Theorem 1, it follows that our algorithms for uncertain edges costs (Theorems 2 and 3 below) imply corresponding algorithms for uncertain activated vertices, and that our lower bounds for uncertain activated vertices (Theorem 4 below) imply corresponding lower bounds for uncertain edge costs.

Theorem 2 (Stochastic optimization upper bound).

- (1) *There is a polynomial-time deterministic algorithm for stochastic matching (OPT_1) that constructs a perfect matching at expected cost is at most $2n^2 \cdot OPT_1$.*
(2) *Given $\beta \in (0, 1)$, there is a polynomial-time randomized algorithm for stochastic matching (OPT_1) that returns a matching whose cardinality, with probability $1 - e^{-n}$ (over the random choices of the algorithm), is at least $(1 - \beta)n$, and whose overall expected cost is $O(OPT_1/\beta)$.*

In particular, for any $\epsilon > 0$ we get a matching of size $(1 - \epsilon)n$ and cost $O(OPT/\epsilon)$ in expectation. Note that by Theorem 4, we have to relax the constraint on the size anyway if we wish to obtain a better-than-log n approximation on the cost, so Part 2 of the Theorem is, in a sense, our best option.

Proof. The proof follows the general paradigm applied to stochastic optimization in recent papers such as [11]: formulate the problem as an integer linear program; solve the linear relaxation and use it to guide the algorithm; and use LP duality (König's theorem, for our problem) for the analysis.

To define the integer program, let X_e indicate whether edge e is bought in the first stage, and for each scenario s , let Z_e^s (resp. Y_e^s) indicate whether edge e is bought in the first stage (resp. in the second stage) and ends up in the perfect matching when scenario s materializes. We obtain:

$$\min \sum_{s \in S} \Pr(s) \left(\sum_e C_e X_e + \sum_e C_e^s Y_e^s \right) \text{ s.t. } \begin{cases} \sum_{e: v \in e} (Z_e^s + Y_e^s) = 1 \quad \forall v \in A \cup B, s \in S \\ Z_e^s \leq X_e \quad \forall e \in E, s \in S \\ X_e, Y_e^s, Z_e^s \in \{0, 1\} \quad \forall e \in E, s \in S. \end{cases}$$

The algorithm solves the standard LP relaxation, in which the last set of constraints is replaced by $0 \leq X_e, Y_e^s, Z_e^s \leq 1$. Let (X_e, Z_e^s, Y_e^s) denote the optimal solution of the LP. Now the proof of the two parts of the theorem diverges:

Proof of part 1. In the first stage, buy every edge e such that $X_e \geq 1/(2n^2)$. In the second stage, under scenario s , buy every edge e such that $Y_e^s \geq 1/(2n^2)$. Finally, output a maximum matching of the set of edges bought. The analysis, which relies on Hall's theorem, is in [13].

Proof of part 2. In the first stage, buy each edge e with probability $1 - e^{-X_e \alpha}$. In the second stage under scenario s , buy each edge e with probability $1 - e^{-Y_e^s \alpha}$, where $\alpha = 8 \ln(2)/\beta$. Finally, output a maximum matching of the set of edges bought. The analysis, which relies on König's theorem, is in [13]. \square

Theorem 3 (Robust optimization). *Given $\beta \in (0, 1)$, there is a polynomial-time randomized algorithm for robust matching (OPT_2) with t scenarios that returns a matching s.t. with probability at least $1 - 2/n$ (over the random choices of the algorithm), the following holds: In every scenario, the algorithm incurs cost $O(OPT_2(1 + \ln(t)/\ln(n))/\beta)$ and outputs a matching of cardinality at least $(1 - \beta)n$.*

Proof. We detail this proof, which is the most interesting one in this section. The integer programming formulation is similar to the one used to prove Theorem 2.

More specifically, let X_e indicate whether edge e is bought in the first stage, and for each scenario s , let Z_e^s (resp. Y_e^s) indicate whether edge e is bought in the first stage (resp. in the second stage) and ends up in the perfect matching when scenario s materializes. We obtain:

$$\min W \text{ s.t. } \begin{cases} \sum_{e:v \in e} (Z_e^s + Y_e^s) = 1 & \forall v \in A \cup B \text{ and } \forall s \in S \\ Z_e^s \leq X_e & \forall e \in E \text{ and } s \in S \\ \sum_e [C_e X_e + C_e^s Y_e^s] \leq W & \forall s \in S \\ X_e, Y_e^s, Z_e^s \in \{0, 1\} & \forall e \in E \text{ and } s \in S. \end{cases}$$

The algorithm solves the standard LP relaxation, in which the last set of constraints is replaced by $0 \leq X_e, Y_e^s, Z_e^s \leq 1$. Let $w, (x_e), (y_e^s), (z_e^s)$ denote the optimal solution of the LP. Let $\alpha = 8 \ln(2)/\beta$ again, and let $T = 3 \ln n$.

- In the first stage, relabel the edges so that $c_1 \geq c_2 \geq \dots$. Let t_1 be maximum such that $x_1 + x_2 + \dots + x_{t_1} \leq T$. For every $j > t_1$, buy edge j with probability $1 - e^{-x_j \alpha}$. (Do not buy any edge $j \leq t_1$.)
- In the second stage, relabel the remaining edges so that $c_1^s \geq c_2^s \geq \dots$. Let t_2 be maximum such that $y_1^s + y_2^s + \dots + y_{t_2}^s \leq T$. For every $j > t_2$, buy edge j with probability $1 - e^{-y_j^s \alpha}$. (Do not buy any edge $j \leq t_2$.)

Finally, the algorithm computes and returns a maximum matching of the set of edges bought.

We note that this construction and the rounding used in the analysis are almost identical to the construction used in strip-packing [14]. The analysis of the cost of the edges bought is the difficult part. We first do a slight change of notations. The cost can be expressed as the sum of at most $2m$ random variables (at most m in each stage). Let $a_1 \geq a_2 \geq \dots$ be the multiset $\{c_e\} \cup \{c_e^s\}$, along with the corresponding probabilities p_i ($p_i = 1 - e^{-x_e \alpha}$ if $a_i = c_e$ is a first-stage cost, and $p_i = 1 - e^{-y_e^s \alpha}$ if $a_i = c_e^s$ is a second-stage cost.) Let X_i be the binary variable with expectation p_i . Clearly, the cost incurred by the algorithm can be bounded above by $X = \sum_{i > t^*} a_i X_i$, where t^* is maximum such that $p_1 + \dots + p_{t^*} \leq T$.

To prove a high-probability bound on X , we will partition $[1, 2m]$ into intervals to define groups. The first group is just $[1, t^*]$, and the subsequent groups are defined in greedy fashion, with group $[j, \ell]$ defined by choosing ℓ maximum so that $\sum_{i \in [j, \ell]} p_i \leq T$. Let G_1, G_2, \dots, G_r be the resulting groups. We have:

$$X \leq \sum_{\ell \geq 2} \sum_{i \in G_\ell} a_i X_i \leq \sum_{\ell \geq 2} \sum_{i \in G_\ell} (\max_{G_\ell} a_i) X_i \leq \sum_{\ell \geq 2} \sum_{i \in G_\ell} (\min_{G_{\ell-1}} a_i) X_i \leq \sum_{\ell \geq 1} (\min_{G_\ell} a_i) \sum_{i \in G_{\ell+1}} X_i.$$

On the other hand, (using the inequality $1 - e^{-Z} \leq Z$), the optimal value OPT^* of the LP relaxation satisfies:

$$\alpha \text{OPT}^* \geq \sum_i a_i p_i \geq \sum_{\ell \geq 1} \sum_{i \in G_\ell} (\min_{G_\ell} a_i) p_i \geq \sum_{\ell \geq 1} (\min_{G_\ell} a_i) (T - 1).$$

It remains, for each group G_ℓ , to apply a standard Chernoff bound to bound the sum of the X_i 's in G_ℓ , and use union bounds to put these results together and yield the statement of the theorem [13]. \square

We note that the proof of Theorem 3 can also be extended to the setting of Theorem 2 to prove a high probability result: For scenario s , with probability at least $1 - 2/n$ over the random choices of the algorithm, the algorithm incurs cost $O(\text{OPT}_s/\beta)$ and outputs a matching of cardinality at least $(1 - \beta)n$, where $\text{OPT}_s = \sum_{E_1} C_e + \sum_{E_2^s} C_e^s$.

Finally, we can show two hardness of approximation results for the explicit scenario case.

Theorem 4 (Stochastic optimization lower bound).

1. *There exists a constant $c > 0$ such that Expression OPT_3 (Eq (3)) is NP-hard to approximate within a factor of $c \ln n$.*
2. *Expression OPT_3 (Eq (3)) is NP-hard to compute, even when there are only two scenarios and τ is bounded.*

Proof. To prove Part 1, we show that when $\tau \geq n^2$, Expression (3) is at least as hard to approximate as Minimum Set Cover: Given a universe $S = \{s_1, \dots, s_n\}$ of elements and a collection $C = \{c_1, \dots, c_k\}$ of subsets of S , find a minimum-cardinality subset \mathcal{SC} of C such that for every $1 \leq i \leq n$, $s_i \in c_j$ for some $c_j \in \mathcal{SC}$. It is known that there exists a constant $c > 0$ such that approximating Minimum Set-Cover to within a factor of $c \ln n$ is NP-hard [18].

Given an instance $(S = \{s_1, \dots, s_n\}; C = \{c_1, \dots, c_k\})$ of Minimum Set-Cover, we construct an instance of the two-stage matching problem with stochastic matching vertices as follows. The graph contains $|S| + 3|C|$ vertices: for every element $s_i \in S$ there is a vertex u_i ; for every set $c_j \in C$, there are three vertices x_j, y_j , and z_j connected by a path $(x_j, y_j), (y_j, z_j)$. For every set c_j and element s_i which belongs to c_j , we have the edge (z_j, u_i) . It is easy to see that the graph is bipartite. The first-stage edge costs are 1 for an (x_i, y_i) edge costs and 0 for the other edges. The second-stage costs are equal to the first-stage costs, multiplied by τ . There are n equally likely second-stage scenarios: In scenario i the vertices in $\{y_1, \dots, y_k\} \cup \{u_i\}$ are active. In [13] we show that the optimal solution to the stochastic matchings instance buys, in the first stage, the edge (x_j, y_j) for each set c_j in some minimum set cover of the input.

The proof of Part 2 is by reduction from the Simultaneous Matchings [7] problem and is also in [13]. □

3 Implicit scenarios

Instead of having an explicit list of scenarios for the second stage, it is common to have instead an implicit description: in the case of uncertain activated vertices, a natural stochastic model is the one in which each vertex is active in the second stage with some probability p , independently of the status of the other nodes. Due to independence, we get that although the total number of possible scenarios can be exponentially large, there is a succinct description consisting of simply specifying the activation probability of each node. In this case, we can no longer be certain that the second-stage graph contains a perfect matching even if the input graph does, so the requirement is, as stated above, to find the largest possible matching. We first prove an interesting lower bound.

3.1 Lower bounds

Theorem 5. *Stochastic optimization with uncertain vertex set is NP-hard to approximate within a certain constant, even with independent vertex activation.*

Proof. We detail this proof, which is the most interesting of our lower bounds. We will use a reduction from Minimum 3-Set-Cover(2), the special case of Minimum Set-Cover where each set has cardinality 3 and each element belongs to two sets [16]. This variant is NP-hard to approximate to within a factor of 100/99 [3]. We will prove that approximating Expression (3) to within a factor of β is at least as hard as approximating 3-set-cover(2) to within a factor of $\gamma = \beta(1 + (3p^2(1-p) + 2p^3)\tau)$. The theorem follows by setting p to be a constant in the interval $[0, 0.0033]$ and $\tau = 1/p$, because then $3p(1-p) + 2p^2 < 1/99$.

Given an instance $(S = \{s_1, \dots, s_n\}; C = \{c_1, \dots, c_k\})$ of 3-set-cover(2), we construct an instance of the two-stage matching problem with uncertain activated vertices as follows (see Figure 2). The graph contains $2|S| + 3|C|$ vertices: for every element $s_i \in S$ there are two vertices u_i, u'_i connected by an edge whose first stage cost is 1; for every set $c_j \in C$, there are three vertices x_j, y_j , and z_j connected by a path $(x_j, y_j), (y_j, z_j)$. For every set c_j and element s_i which belongs to c_j , we have the edge (z_j, u_i) . It is easy to see that the graph is bipartite. The first-stage edge costs are 1 for an (x_i, y_i) edge and 0 for the other edges. The second-stage costs are equal to the first-stage costs, multiplied by τ . In the second-stage scenarios, each vertex u_i is active with probability p and each y_i is active with probability 1.

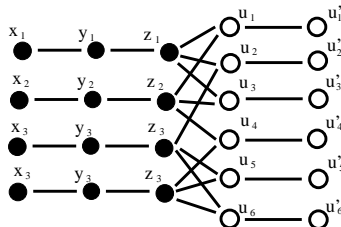


Fig. 2. The graph obtained from the 3-Set-Cover(2) instance $\{s_1, s_2, s_3\}, \{s_1, s_3, s_4\}, \{s_2, s_5, s_6\}, \{s_4, s_5, s_6\}$.

If $p > 1/\tau$, then buying all (u_i, u'_i) edges in the first stage at cost n is optimal. To see why, assume that an algorithm spends $n' < n$ in the first stage. In the second stage, the expected number of active vertices that cannot be matched is at least $(n-n')p$ and the expected cost of matching them is $\tau(n-n')p > (n-n')$. We assume in the following that $p \leq 1/\tau$.

Consider a minimum set cover \mathcal{SC} of the input instance. Assume that in the first stage we buy (at cost 1) the edge (x_j, y_j) for every set $c_j \in \mathcal{SC}$. In the second stage, let I be the set of active vertices and find, in a way to be

described shortly, a matching M_I between a subset I' of I and the vertex-set $\{z_j : c_j \in \mathcal{SC}\}$, using (z_j, u_i) -edges from the graph. Buy the edges in M_I (at cost 0). For every $i \in I \setminus I'$, buy the edge (u_i, u'_i) at cost τ . Now, all active u_i vertices are matched, and it remains to ensure that the y -vertices are matched as well. Assume that y_j is unmatched. If z_j is matched with some u_i node, this is because $c_j \in \mathcal{SC}$, so we bought the edge (x_j, y_j) in the first stage and can now use it at no additional cost. Otherwise, we buy the edge (y_j, z_j) at cost 0. The second stage has cost equal to τ times the cardinality of $I \setminus I'$ and the first stage has cost equal to the cardinality of the set cover. The matching M_I is found in a straightforward manner: Given \mathcal{SC} , each element chooses exactly one set among the sets covering it, and, if it turns out to be active, will only try to be matched to that set. Each set in the set cover will be matched with one element, chosen arbitrarily among the active vertices who try to be matched with it.

To calculate the expected cost of matching the vertices of $I - I'$, consider a set in \mathcal{SC} . It has 3 elements, and is chosen by at most 3 of them. Assume that it is chosen by all 3. With probability $(1 - p)^3 + 3p(1 - p)^2$, at most one of them is active and no cost is incurred in the second stage. With probability $3p^2(1 - p)$, two of them are active and a cost of τ is incurred. With probability p^3 , all three of them are active and a cost of 2τ is incurred, for an expected cost of $(3p^2(1 - p) + 2p^3)\tau$. If the set is chosen by two elements, the expected cost is at most $p^2\tau$, and if it is chosen by fewer, the expected cost is 0. Thus in all cases the expected cost of matching $I \setminus I'$ is bounded by $|\mathcal{SC}|(3p^2(1 - p) + 2p^3)\tau$. With a cost of $|\mathcal{SC}|$ for the first stage, we get that the total cost of the solution is at most $|\mathcal{SC}|(1 + (3p^2(1 - p) + 2p^3)\tau)$.

On the other hand, let \mathcal{M}_1 be the set of cost-1 edges bought in the first stage. Let an (x_i, y_i) edge represent the set c_i and let a (u_i, u'_i) edge represent the singleton set $\{s_i\}$. Now, assume that \mathcal{M}_1 does not correspond to a set cover of the input instance. Let x be the number of elements which are not covered by the sets corresponding to \mathcal{M}_1 and let X be the number of active elements among those x . In the second stage, the algorithm will have to match each uncovered element vertex u_i , either by its (u_i, u'_i) edge (at cost n) or by a (z_j, u_i) edge for some set c_j where $s_i \in c_j$. In the latter case, it would have to buy the edge (x_i, y_i) , again at cost n . The second stage cost, therefore, is at least Xn . But the expected value of X is x/n , thus the total expected cost is at least $|\mathcal{M}_1| + x$. Since we could complete \mathcal{M}_1 into a set cover by adding at most one set per uncovered element, we have $x + |\mathcal{M}_1| \geq |\mathcal{SC}|$.

In summary, we get that Expression (3) satisfies

$$|\mathcal{SC}| \leq \text{OPT} \leq |\mathcal{SC}|(1 + (3p^2(1 - p) + 2p^3)\tau).$$

This means that if we can approximate our problem within a factor of β , then we can approximate Minimum 3-Set-Cover(2) within a factor of $\gamma = \beta(1 + (3p^2(1 - p) + 2p^3)\tau)$, and the theorem follows. \square

Using similar ideas, we prove the following related result in [13].

Theorem 6. *The case of uncertain, independent, edge costs is NP-hard to approximate within a certain constant.*

3.2 Upper bound in a special case

We can show that when $c_e = 1$ for all $e \in E$, it is possible to construct a perfect matching cheaply when the graph has certain properties. We study the case in which B is significantly larger than A .

Theorem 7. *Assume that the graph contains n vertex-disjoint stars s_1, \dots, s_n such that star s_i contains $d = \max\{1, \ln(\tau p)\} / \ln(1/(1-p)) + 1$ vertices from B and is centered at some vertex of A . Then there is an algorithm whose running time is polynomial in n and which returns a maximum-cardinality matching of the second stage graph, whose expected cost is $O(OPT_3 \cdot \min\{1, \ln(\tau p)\})$.*

To prove this, let $A = \{a_1, \dots, a_n\}$ and $B = \{b_1, \dots, b_m\}$. Let E_1 be the edges in the stars. Let B_2 be the vertices which are active in the second stage. Here is the algorithm. In the first stage, if $\tau p \leq e$ then the algorithm buys nothing; else, the algorithm buys all edges of E_1 , paying nd . In the second stage, the algorithm completes its set of edges into a perfect matching in the cheapest way possible. It remains to show that the expected cost of the second stage is low, compared to the optimal cost. We do this by showing that the number of edges bought in the second stage is proportional to the number of nodes of A that have at most one active node in their stars, and that there are few such nodes. The details are in [13].

3.3 Generalization: The Black Box Model

With independently activated vertices, the number of scenarios is extremely large, and so solving an LP of the kind described in previous sections is prohibitively time-consuming. However, in such a situation there is often a *black box* sampling procedure that provides, in polynomial time, an unbiased sample of scenarios; then one can use the SAA method to simulate the explicit scenarios case, and, if the edge cost distributions have bounded second moment, one can extend the analysis so as to obtain a similar approximation guarantee. The main observation is that the value of the LP defined by taking a polynomial number of samples of scenarios tightly approximates the the value of the LP defined by taking all possible scenarios. An analysis similar to [5] gives:

Theorem 8. *Consider a two-stage edge stochastic matching problem with (1) a polynomial time unbiased sampling procedure and (2) edge cost distributions have bounded second moment. For any constants $\epsilon > 0$ and $\delta, \beta \in (0, 1)$, there is a polynomial-time randomized algorithm that outputs a matching whose cardinality is at least $(1-\beta)n$ and, with probability at least $1-\delta$ (over the choices of the black box and of the algorithm), incurs expected cost $O(OPT/\beta)$ (where the expectation is over the space of scenarios).*

References

1. J. Birge and F. Louveaux. *Introduction to Stochastic Programming*. Springer, 1997.

2. M. Charikar, C. Chekuri, and M. Pal. Sampling bounds for stochastic optimization. In *APPROX-RANDOM*, pages 257–269, 2005.
3. M. Chlebík and J. Chlebíková. Inapproximability results for bounded variants of optimization problems. In *FCT*, volume 2751 of *LNCS*, pages 27–38, 2003.
4. K. Dhamdhere, V. Goyal, R. Ravi, and M. Singh. How to pay, come what may: Approximation algorithms for demand-robust covering problems. In *FOCS*, pages 367–378, 2005.
5. K. Dhamdhere, R. Ravi, and M. Singh. On two-stage stochastic minimum spanning trees. In *IPCO*, volume 3509 of *LNCS*, pages 321–334, 2005.
6. S. Dye, L. Stougie, and A. Tomasgard. The stochastic single resource service-provision problem. *Naval Research Logistics*, 50:257–269, 2003.
7. K.M. Elbassioni, I. Katriel, M. Kutz, and M. Mahajan. Simultaneous matchings. In *ISAAC*, volume 3827 of *LNCS*, pages 106–115, 2005.
8. A.D. Flaxman, A.M. Frieze, and M. Krivelevich. On the random 2-stage minimum spanning tree. In *SODA*, pages 919–926, 2005.
9. A. Gupta and M. Pál. Stochastic steiner trees without a root. In *ICALP*, volume 3580 of *LNCS*, pages 1051–1063, 2005.
10. A. Gupta, M. Pál, R. Ravi, and A. Sinha. Boosted sampling: approximation algorithms for stochastic optimization. In *STOC*, pages 417–426. ACM, 2004.
11. A. Gupta, R. Ravi, and A. Sinha. An edge in time saves nine: LP rounding approximation algorithms for stochastic network design. In *FOCS*, pages 218–227, 2004.
12. N. Immerlica, D. Karger, M. Minkoff, and V.S. Mirrokni. On the costs and benefits of procrastination: approximation algorithms for stochastic combinatorial optimization problems. In *SODA*, pages 691–700, 2004.
13. I. Katriel, C. Kenyon-Mathieu, and E. Upfal. Commitment under uncertainty: Two-stage stochastic matching problems, 2007. ECCO <http://eccc.hpi-web.de/eccc/>.
14. C. Kenyon and E. Rémila. A near-optimal solution to a two-dimensional cutting stock problem. *Math. Oper. Res.*, 25(4):645–656, 2000.
15. N. Kong and A.J. Schaefer. A factor 1/2 approximation algorithm for two-stage stochastic matching problems. *Eur. J. of Operational Research*, 172:740–746, 2006.
16. C.H. Papadimitriou and M. Yannakakis. Optimization, approximation, and complexity classes. *J. of Computing and System Sciences*, 43:425–440, 1991.
17. R. Ravi and A. Sinha. Hedging uncertainty: Approximation algorithms for stochastic optimization problems. In *IPCO*, volume 3064 of *LNCS*, pages 101–115, 2004.
18. R. Raz and S. Safra. A sub-constant error-prob. low-degree test, and a sub-constant error-prob. PCP characterization of NP. In *STOC*, pages 475–484, 1997.
19. D.B. Shmoys and M. Sozio. Approximation algorithms for 2-stage stochastic scheduling problems. In *IPCO*, 2007.
20. D.B. Shmoys and C. Swamy. The sample average approximation method for 2-stage stochastic optimization, 2004.
21. D.B. Shmoys and C. Swamy. Stochastic optimization is almost as easy as deterministic optimization. In *FOCS*, pages 228–237, 2004.
22. C. Swamy and D.B. Shmoys. The sampling-based approximation algorithms for multi-stage stochastic optimization. In *FOCS*, pages 357–366, 2005.
23. C. Swamy and D.B. Shmoys. Algorithms column: Approximation algorithms for 2-stage stochastic optimization problems. *ACM SIGACT News*, 37(1):33–46, 2006.
24. B. Verweij, S. Ahmed, A.J. Kleywegt, G. Nemhauser, and A. Shapiro. The sample average approximation method applied to stochastic routing problems: a computational study. *Comp. Optimization and Applications*, 24:289–333, 2003.